



AGH

**AKADEMIA GÓRNICZO-HUTNICZA
IM. STANISŁAWA STASZICA
W KRAKOWIE**

Metody Obliczeniowe w Nauce i Technice

Sprawozdanie z laboratorium 5 – Aproksymacja
wielomianami algebraicznymi

Michał Szafarczyk

gr. Śr. 17:50 – 19:20

Narzędzia i sprzęt wykorzystany do zrealizowania ćwiczenia

Komputer z systemem Windows 10 x64 Home

Procesor: Intel Core i7-10750H @2.60 GHz / 5.00 GHz

Pamięć RAM: 32 GB

Język: Python 3.9

Środowisko: PyCharm

Użyte biblioteki pythonowskie:

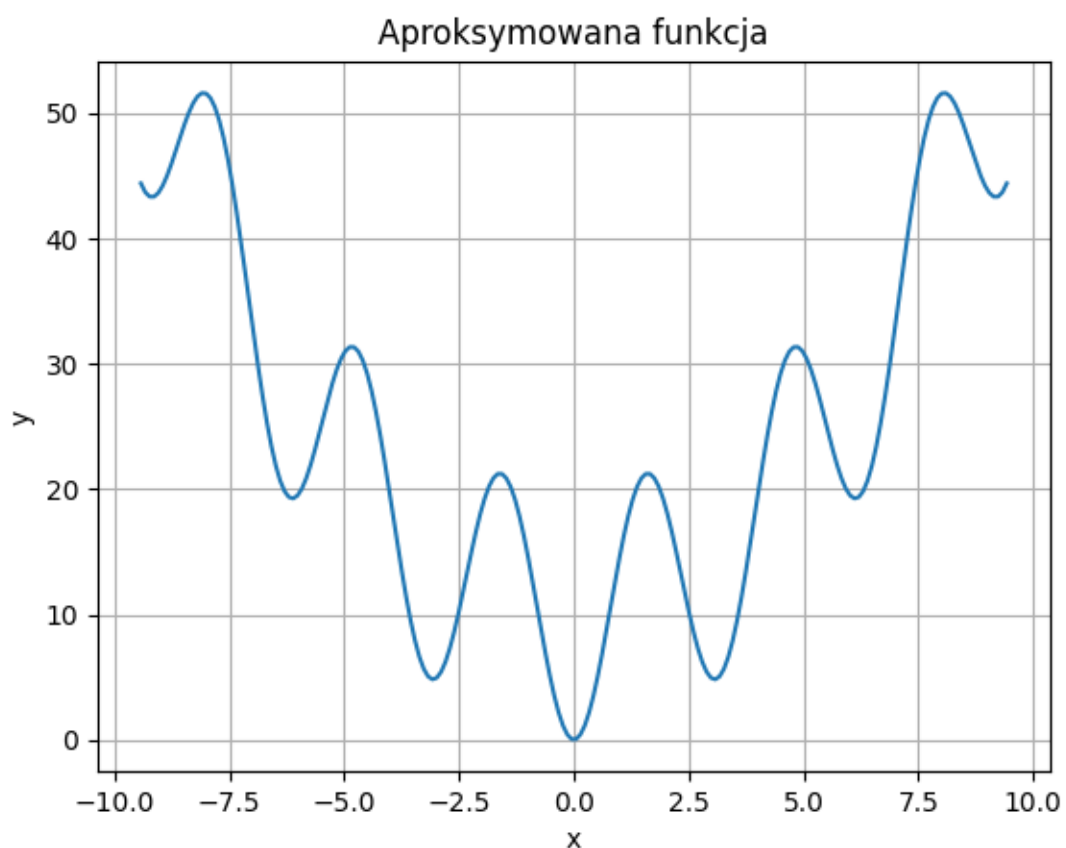
- Numpy – do wykonywania różnych operacji na liczbach
- Matplotlib – dla rysowania wykresów

1. Zadana funkcja:

Przekazana wraz z zadaniem funkcja, którą będziemy interpolować, dana jest wzorem

$$f(x) = 10 \cdot m + \frac{x^2}{k} - 10 \cdot m \cdot \cos(k \cdot x), \quad x \in [-3\pi, 3\pi]$$

gdzie $m = 1$, $k = 2$



Wykres 1.1 – Aproksymowana funkcja

2. Aproksymacja za pomocą wielomianów jako funkcji bazowych:

Błąd średniokwadratowy:

Rozpatrujemy przybliżenie aproksymacyjne:

$$F(x) = c_0\varphi_0 + c_1\varphi_1 + c_2\varphi_2 + \dots + c_m\varphi_m = \sum_{j=0}^m c_j\varphi_j \quad (2.1)$$

Jest to inaczej kombinacja liniowa przyjętych funkcji bazowych $\varphi_j(x)$. Współczynniki c_j nie są początkowo znane. Wyliczenie funkcji aproksymującej będzie polegało na wyznaczeniu tych współczynników. Spośród nieskończenie wielu rozwiązań dla współczynników poszukujemy rozwiązania, dla którego funkcja aproksymująca ma minimalną normę średniokwadratową, przedstawiającą się równaniem:

$$E = \sum_{j=0}^n w(x_j) \left(f(x_j) - F(x_j) \right)^2 \quad (2.2)$$

Gdzie $w(x)$ jest funkcją wagową i $w(x) \geq 0$.

Układ równań normalnych:

Używając wzorów (2.1) oraz (2.2) otrzymujemy:

$$E = \sum_{j=0}^n w(x_j) \left[f(x_j) - \sum_{i=0}^m c_i\varphi_i(x_j) \right]^2 \quad (2.3)$$

Ponieważ szukamy c_i , dla których E będzie minimalna, potrzebujemy wszystkie pochodne cząstkowe, gdzie $\frac{\partial E}{\partial c_k} = 0$, $k = 0, 1, 2, \dots, m$. Pochodne będą przyjmowały postać:

$$\frac{\partial E}{\partial c_k} = \sum_{j=0}^n w(x_j) \left[f(x_j) - \sum_{i=0}^m c_i\varphi_i(x_j) \right] \varphi_k(x_j) \quad (2.4)$$

Przyrównując równania kolejne pochodne do 0 i używając danych punktów x_j , na podstawie których będziemy dokonywać aproksymacji. Utworzony układ jest układem nadokreślonym, to znaczy otrzymujemy $n + 1$ równań, ale potrzebujemy wyznaczyć $m + 1$ niewiadomych, przy $m \leq n$. Po rozwiązaniu tego układu otrzymamy funkcję aproksymującą, dla której norma dana wzorem (2.2) jest minimalna.

Aproksymacja wielomianami algebraicznymi:

Funkcjami bazowymi dla aproksymacji będą wielomiany w postaci:

$$\varphi_j(x) = x^j, \quad j \in \{0, 1, 2, \dots, m\} \quad (2.5)$$

Funkcja aproksymująca będzie miała postać:

$$f(x) = \sum_{j=0}^m c_j \varphi_j(x) = \sum_{j=0}^m c_j x^j \quad (2.6)$$

Układ normalny z (2.4) przyjmuje więc postać

$$\begin{aligned} \sum_{j=0}^n w(x_j) \left[F(x_j) - \sum_{i=0}^m c_i x_j^i \right] x_j^k &= 0 \\ \sum_{j=0}^n w(x_j) x_j^k \sum_{i=0}^m c_i x_j^i &= \sum_{j=0}^n w(x_j) F(x_j) x_j^k \end{aligned}$$

gdzie $k = 0, 1, 2, \dots, m$

$$\sum_{i=0}^m \left(\sum_{j=0}^n w(x_j) x_j^{i+k} \right) c_i = \sum_{j=0}^n w(x_j) F(x_j) x_j^k \quad (2.7)$$

Wzór (2.7) można zapisać w postaci macierzowej:

$$\begin{pmatrix} \sum w_j & \sum w_j x_j & \sum w_j x_j^2 & \cdots & \sum w_j x_j^m \\ \sum w_j x_j & \sum w_j x_j^2 & \sum w_j x_j^3 & \cdots & \sum w_j x_j^{m+1} \\ \sum w_j x_j^2 & \cdots & \ddots & \ddots & \vdots \\ \vdots & \cdots & \ddots & \ddots & \vdots \\ \sum w_j x_j^m & \sum w_j x_j^{m+1} & \sum w_j x_j^{m+2} & \cdots & \sum w_j x_j^{2m} \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_m \end{pmatrix} = \begin{pmatrix} \sum w_j F_j \\ \sum w_j F_j x_j \\ \vdots \\ \sum w_j F_j x_j^m \end{pmatrix} \quad (2.8)$$

która posłuży nam do obliczenia współczynników c_j w programie.

Dla uproszczenia oznaczymy macierze w tym równaniu jako:

$$G \cdot A = B \quad (2.9)$$

3. Wyznaczenie błędów:

Miarami, którą przyjmujemy dla badania jak dokładne względem oryginalnej funkcji są funkcje sklepane będą:

- Maksimum z różnic wartości pomiędzy funkcją interpolowaną, a funkcją sklepaną. Liczymy je według wzoru:

$$\max(|f(x) - S(x)|)$$

- Suma kwadratów różnic wartości funkcji interpolowanej i funkcji sklepanej, liczonej jako:

$$\sum_{i=0}^n [f(x) - S(x)]^2$$

Oczywiście nie jesteśmy w stanie dokładnie zbadać całego badanego obszaru zadanej funkcji pisząc program, dlatego różnice wartości będziemy badać w przyjętej liczbie punktów. Jak określiliśmy w punkcie 1., zadaną do testów funkcję interpolujemy w przedziale $x \in [-3\pi, 3\pi]$. Dla takiego przedziału postanowiliśmy przyjąć 500 równoodległych punktów dla badania błędów pomiędzy funkcjami (W kodzie liczba punktów została zdefiniowana jako wartość globalna, którą można zmienić w dowolnym momencie, jeśli przedział byłby większy).

4. Testy:

Testy przeprowadzać będziemy dla następujących m (ilość funkcji bazowych) oraz n danych punktów dla aproksymacji:

$$m \in \{2, 3, 5, 7, 10, 12, 15\}$$
$$n \in \{5, 7, 10, 12, 15, 25, 30, 50, 70, 100, 120, 150\}$$

Oczywiście testy przeprowadzane będą jedynie dla kombinacji gdzie $m \leq n$. Wartość funkcji $w(x)$ dla każdego x przyjmujemy jako 1, ponieważ chcemy, aby wszystkie punkty były równie istotne podczas aproksymacji.

4.2. Węzły równoodległe:

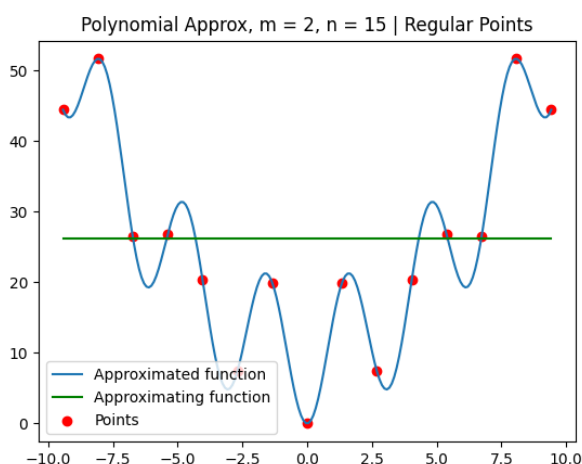
$m \backslash n$	5	7	10	12	15	25	30	50	70	100	120	150
2	92.741	151.209	89.854	89.285	88.695	87.71	87.456	86.939	86.714	86.544	86.478	86.411
3	99.905	139.999	90.991	87.331	84.547	80.059	78.824	76.155	74.919	73.951	73.565	73.174
5	184.58	139.999	96.68	86.73	82.644	79.917	79.457	78.54	78.112	77.766	77.624	77.479
7	-	139.999	93.54	87.257	82.511	79.496	79.302	79.21	79.226	79.238	79.241	79.241
10	-	-	203.153	132.165	85.144	76.996	76.553	76.782	77.234	77.686	77.885	78.096
12	-	-	-	641.951	194.522	76.669	77.518	77.878	77.646	77.34	77.19	77.024
15	-	-	-	-	12666.816	202.16	102.335	77.448	77.979	77.927	77.81	77.639
30	-	-	-	-	-	-	1427.8	16.366	3.369	0.937	1.857	0.768

Tabela 4.1 – Tabela Największych bezwzględnych błędów dla aproksymacji dla węzłów równoodległych

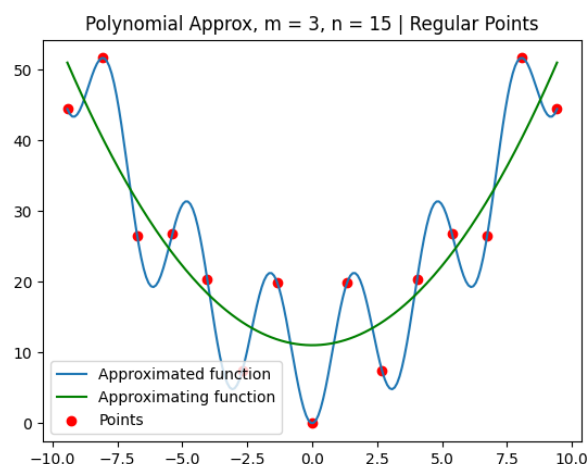
$m \backslash n$	5	7	10	12	15	25	30	50	70	100	120	150
2	49860	49568	45195	44854	44599	44343	44304	44253	44242	44237	44236	44235
3	12011	29850	10709	10434	10288	10128	10098	10052	10040	10034	10033	10032
5	28439	29850	10660	10150	10016	9926	9908	9875	9864	9858	9857	9856
7	-	29850	10135	9607	9249	9182	9180	9179	9179	9179	9179	9179
10	-	-	41749	13770	9387	8514	8473	8386	8349	8328	8322	8318
12	-	-	-	148426	18004	8519	8467	8391	8352	8325	8317	8311
15	-	-	-	-	39522910	12249	7410	6477	6438	6410	6400	6391
30	-	-	-	-	-	-	7369042	843.30	22.505	4.656	55.367	12.195

Tabela 4.2 – Tabela Sum kwadratów różnic dla węzłów równoodległych

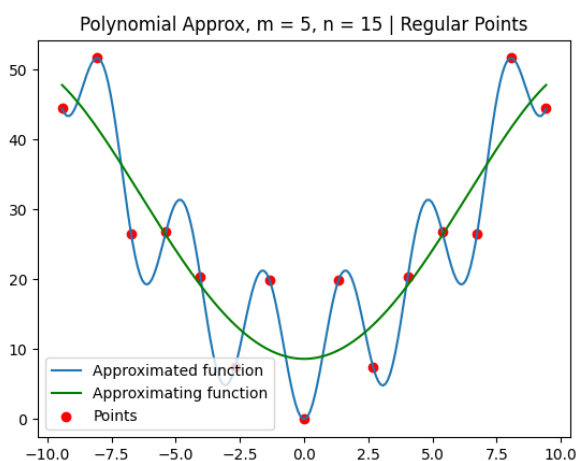
Jak widzimy na podstawie tabel 4.1 oraz 4.2 jedynie dla kilku pierwszych wartości n funkcje aproksymujące są istotnie różne. Dla większych wartości te różnice stają się praktycznie niezauważalne. Jeżeli chodzi o wartości m , to łatwo zauważyć, że mają one znacznie większy wpływ na dokładność wynikowej funkcji. Zobrazujemy kilka przykładów testów:



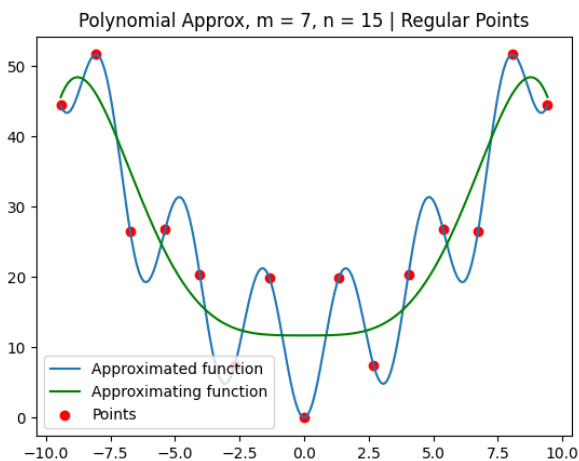
Wykres 4.1.1 dla $m=2$, $n=15$



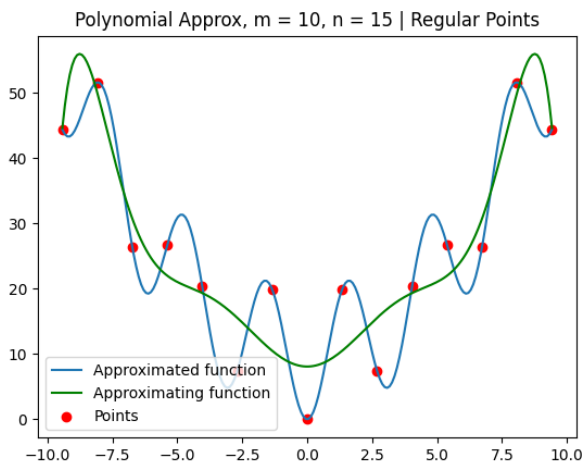
Wykres 4.1.2 dla $m=3$, $n=15$



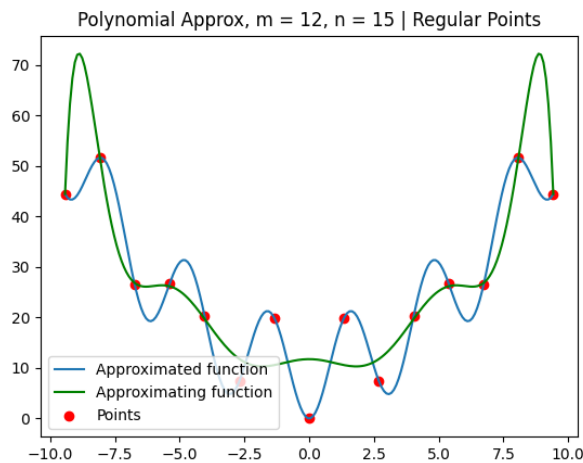
Wykres 4.1.3 dla $m=5$, $n=15$



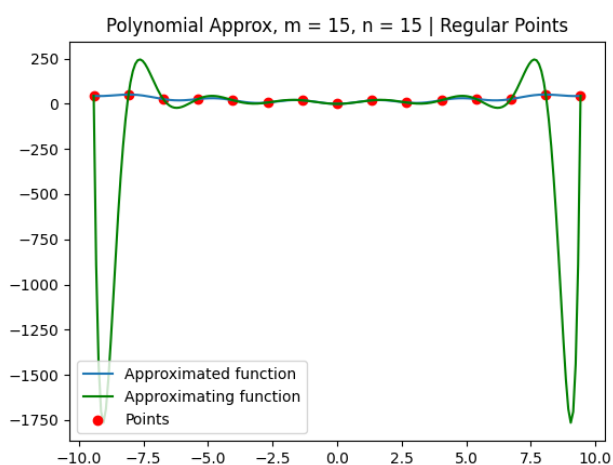
Wykres 4.1.4 dla $m=7$, $n=15$



Wykres 4.1.5 dla $m=10$, $n=15$

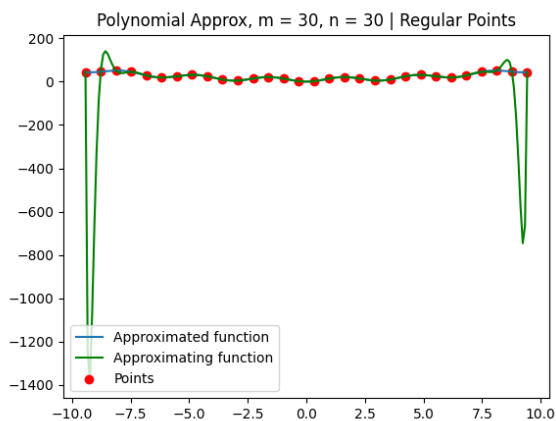


Wykres 4.1.6 dla $m=12$, $n=15$



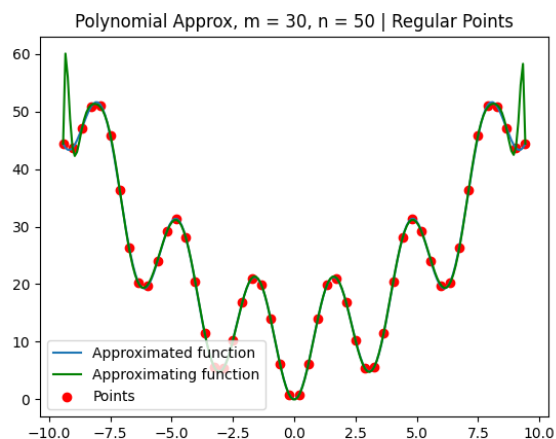
Wykres 4.1.7 dla $m=15$, $n=15$

Zaczynając od stopnia 1 wielomianu na wykresie 4.1, funkcja aproksymująca, wraz ze wzrastającym m , zbliża się do funkcji aproksymowanej. Słabą stroną aproksymacji wielomianami na węzłach równoodległych jest występowanie efektu Rungego, który możemy zaobserwować już dla $m = 10$. Dla $m = 15$ efekt ten jest już bardzo wyraźnie widoczny i dla większych m wykresy będą wyglądały podobnie. Np. dla $m = 30$, $n = 30$:



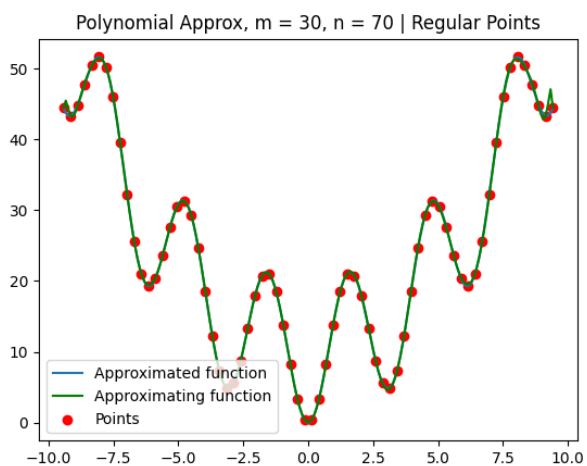
Wykres 4.1.8 dla $m=30$, $n=30$

Jednakże n ma również znaczenie dla występowania tego efektu oraz jego skali. Przykładowo pozostając przy $m = 30$, ale biorąc tym razem $n = 50$ otrzymamy dużo dokładniejszą funkcję:

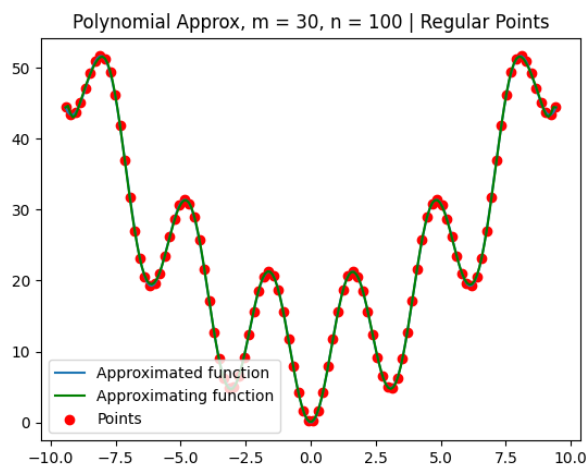


Wykres 4.1.9 dla $m=30$, $n=50$

Dla większych n efekt ten staje się nieistotny w skali całej funkcji:



Wykres 4.1.10 dla $m=30$, $n=70$



Wykres 4.1.11 dla $m=30$, $n=100$

4.2. Węzły Czebyszewa:

Na poprzednich laboratoriach używaliśmy węzłów Czebyszewa aby przeciwdziałać efekcie Rungego. Użyjemy ich również do testów przy aproksymacji, aby sprawdzić jaki wpływ będzie miał wybór węzłów na dokładność aproksymacji.

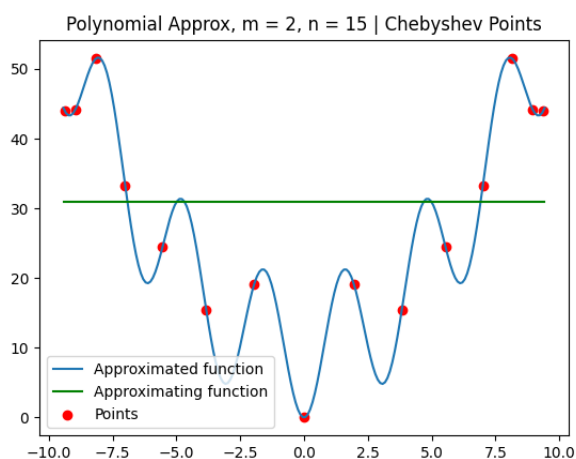
m\	n	5	7	10	12	15	25	30	50	70	100	120	150
2		27.41	30.712	32.955	31.035	30.869	30.87	30.87	30.87	30.87	30.87	30.87	30.87
3		15.032	12.263	18.612	12.52	12.393	12.395	12.395	12.395	12.395	12.395	12.395	12.395
5		18.923	17.905	20.157	11.6	12.429	12.4	12.4	12.4	12.4	12.4	12.4	12.4
7		-	18.332	24.059	19.061	11.681	11.858	11.858	11.858	11.858	11.858	11.858	11.858
10		-	-	21.405	21.599	11.293	10.67	10.67	10.67	10.67	10.67	10.67	10.67
12		-	-	-	20.379	14.621	11.914	11.914	11.914	11.914	11.914	11.914	11.914
15		-	-	-	-	15.354	12.099	12.099	12.099	12.099	12.099	12.099	12.099
30		-	-	-	-	-	-	0.925	0.337	0.251	1.564	0.272	0.329

Tabela 4.3 – Tabela Największych bezwzględnych błędów dla aproksymacji dla węzłów Czebyszewa

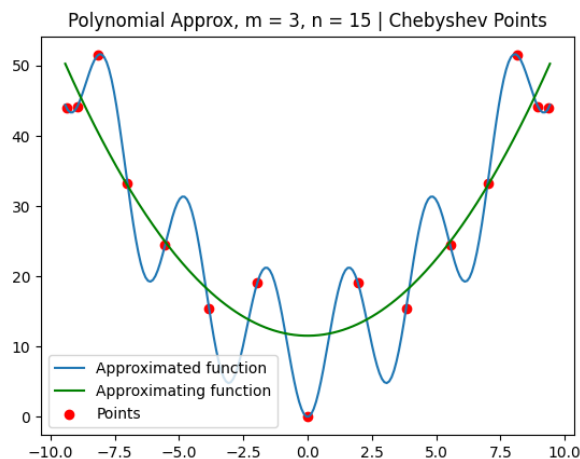
m\	n	5	7	10	12	15	25	30	50	70	100	120	150
2		45538	51092	57348	51868	51464	51466	51466	51466	51466	51466	51466	51466
3		14681	10327	16302	10717	10453	10454	10454	10454	10454	10454	10454	10454
5		15676	15422	17657	10256	10180	10172	10172	10172	10172	10172	10172	10172
7		-	16966	17835	12753	9217	9233	9233	9233	9233	9233	9233	9233
10		-	-	18396	15570	8366	8394	8394	8394	8394	8394	8394	8394
12		-	-	-	15367	9647	8557	8557	8557	8557	8557	8557	8557
15		-	-	-	-	12114	6632	6632	6632	6632	6632	6632	6632
30		-	-	-	-	-	-	52.01	7.404	3.766	136.69	2.864	5.805

Tabela 4.2 – Tabela Sum kwadratów różnic dla węzłów Czebyszewa

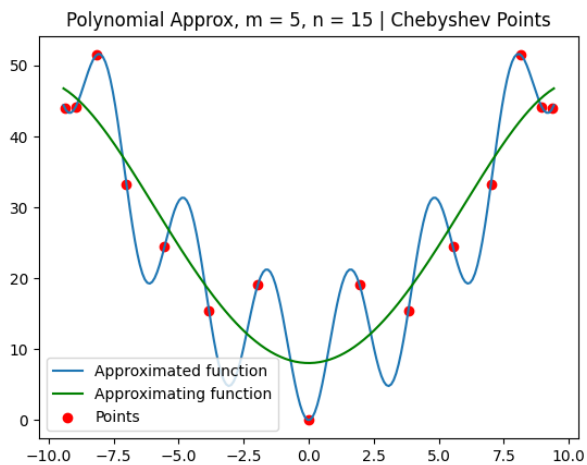
Na pierwszy rzut oka jest widoczne, że zastosowanie węzłów Czebyszewa daje znacznie lepsze wyniki w przypadku obu błędów. Jak zobaczymy zaraz na wykresach, niwelowany jest również efekt Rungego.



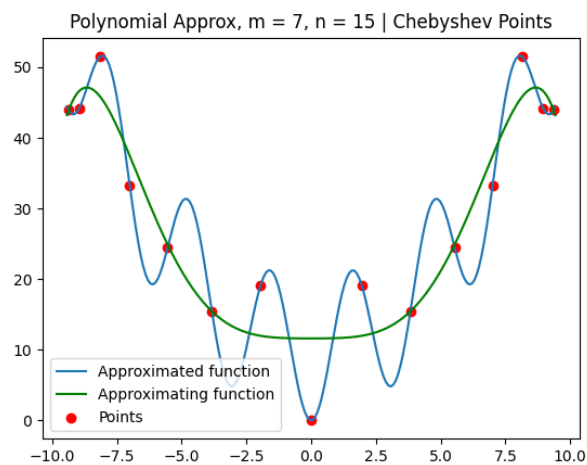
Wykres 4.2.1 dla m=2, n=15



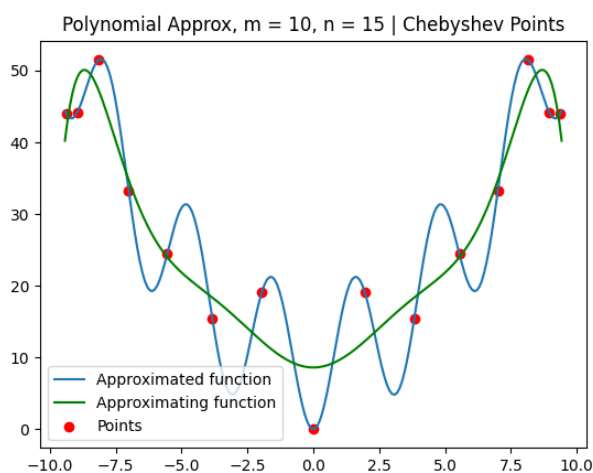
Wykres 4.2.2 dla m=3, n=15



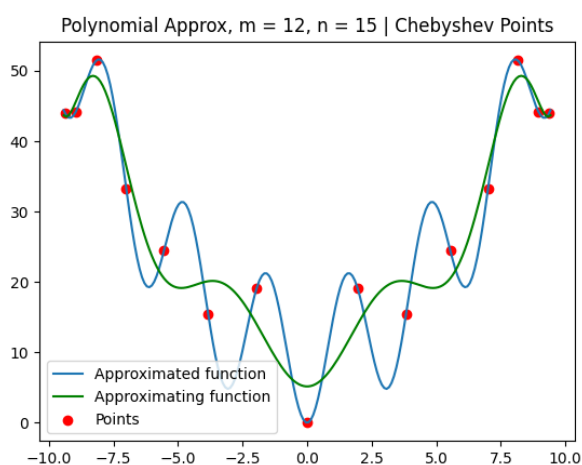
Wykres 4.2.3 dla $m=5$, $n=15$



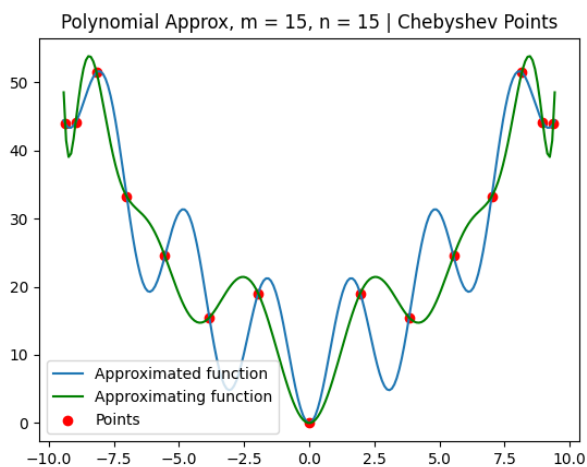
Wykres 4.2.4 dla $m=7$, $n=15$



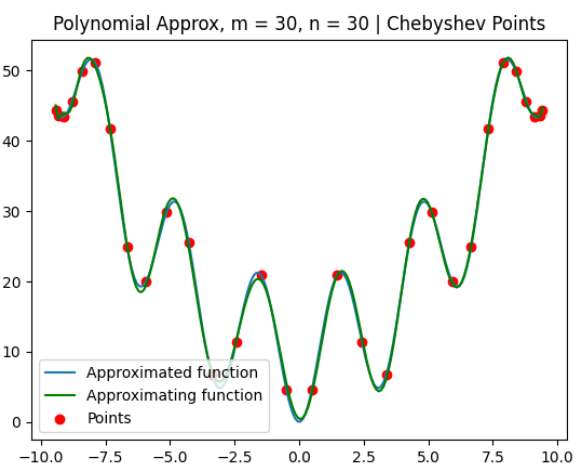
Wykres 4.2.5 dla $m=10$, $n=15$



Wykres 4.2.6 dla $m=12$, $n=15$



Wykres 4.2.7 dla $m=15$, $n=15$



Wykres 4.2.8 dla $m=30$, $n=30$

Porównując wykresy 4.1.6 – 4.1.8 z wykresami 4.2.6 – 4.2.8 widzimy znaczącą zmianę w zachowaniu aproksymującej funkcji na krańcach przedziału. Funkcja na wykresie 4.2.8 jest dobrym przybliżeniem oryginalnej funkcji, tymczasem na wykresie 4.1.8 występują duże błędy.

5. Podsumowanie:

Aproksymacja wielomianami algebraicznymi daje nam prostą funkcję aproksymującą. Nie jest to metoda odporna na efekt Rungego i nie jest najefektywniejsza. W większości przypadków jednak, może być ona wystarczająca – np. jeżeli nie potrzebujemy używać dużych wartości m . Dobrym ulepszeniem jest wykorzystanie węzłów Czebyszewa – jak pokazaliśmy w punkcie 4. niweluje to efekt Rungego i daje lepsze przybliżenie oryginalnej funkcji.

Dla poprawienia efektywności obliczeń powinniśmy korzystać z wielomianów ortogonalnych, dla których znika złe uwarunkowanie zadania dzięki diagonalności macierzy we wzorach (2.8), (2.9).